สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



**รายงานผลการวิจัยประจำปีงบประมาณ 2541**

**เสนอคณะกรรมการวิจัยแห่งชาติ**

**เรื่อง**

# การออกแบบโปรแกรมโฮสคอมพิวเตอร์กับเครื่องควบคุมแบบตรรก จำนวนหลายจุด

# Support Software for Multiunit Programmable Logic Controllers

RCH
QA
76.63
ส828ร
เลขหมู่.....
เลขทะเบียน 40399
วัน, เดือน, ปี 14 ก.ย. 2544
.b 11105777x
.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# รายงานผลการวิจัยประจำปีงบประมาณ 2541
# เสนอคณะกรรมการวิจัยแห่งชาติ

## เรื่อง

# การออกแบบโปรแกรมโฮสคอมพิวเตอร์กับเครื่องควบคุมแบบตรรก จำนวนหลายจุด

# Support Software for Multiunit Programmable Logic Controllers


**นายสุพรรณ กุลพาณิชย์** **หัวหน้าโครงการ**

**ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม**

**คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**



**ขณะนี้ได้ทำบทความขึ้น 1 ฉบับ เรียบร้อยแล้ว โดยใช้ชื่อว่า "การออกแบบโปรแกรมโฮส-คอมพิวเตอร์กับเครื่องควบคุมแบบตรรกจำนวนหลายจุด" ในวารสารการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 19 วันที่ 7-8 พฤศจิกายน 2539**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# การออกแบบโปรแกรมโฮสคอมพิวเตอร์ กับเครื่องควบคุมแบบตรรกจำนวนหลายจุด

# (SUPPORT SOFTWARE FOR MULTIUNIT PROGRAMMABLE LOGIC CONTROLLERS )

ทวีพล ซื่อสัตย์ * วิริยะ กองรัตน์ ** ผศ.วิทยา ทิพย์สุวรรณพร** ผศ.สุพรรณ กุลพาณิชย์**

**บทคัดย่อ**

บทความนี้กล่าวถึงโปรแกรมการทำงานบนเครื่องคอมพิวเตอร์ใช้เป็นระบบจัดการในรูปแบบของโฮสคอมพิวเตอร์เพื่อต่อร่วมกับเครื่องควบคุมแบบตรรกที่โปรแกรมได้ จำนวนหลายชุดซึ่งฮาร์ดแวร์ได้ถูกพัฒนาขึ้นมาและเผยแพร่ในเอกสารอ้างอิงหมายเลข (1) โดยผ่านระบบสื่อสารข้อมูลตามมาตราฐาน RS 485 ที่ต่อได้ทั้งแบบจุดต่อจุด และแบบแยกกระจาย จุดประสงค์เพื่อทำการตรวจสอบสภาวะการทำงานของหน่วยความจำต่าง ๆ เพื่อช่วยในการติดตามผล หรือการอ่านเขียนข้อมูล เพื่อจัดเก็บลงในแฟ้มข้อมูล โปรแกรมที่ได้ออกแบบนี้ถูกพัฒนาโดย Visual Basic ที่ทำงานบน Microsoft Windows และมีฟังก์ชั่น อื่น ๆ ที่จำเป็น เพื่ออำนวยความสะดวกในการใช้งาน ที่กล่าวมานี้ จะสามารถกระทำบนเครื่องควบคุมที่ต่อร่วมอยู่ภายใต้ ระบบที่ได้ออกแบบไว้

**Abstract**

This paper present , soft ware for manage system in host computer . The computer have connect to the programmable logic control , through RS 485 serial communication bus , there are two methord for connecting system are point to point and multidrop

This software use for monitor , write data or upload and download boolean program to save in files which develope by Microsoft Visual Basic for Windows. Purpose for easeful operation with PLC

---

* นักศึกษา ปริญญาโท วิศวกรรมไฟฟ้า สจล.

** อาจารย์ประจำคณะวิศวกรรมศาสตร์

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม สจล.

---

## 1.บทนำ

ปัจจุบัน เทคโนโลยีทางด้านอุตสาหกรรม มีการพัฒนาและเปลี่ยนแปลงไปในทางที่สูง มากขึ้น ดังนั้นระบบควบคุมที่มีประสิทธิภาพโดยเฉพาะอย่างยิ่ง ระบบควบคุมสำหรับโรงงานอัตโนมัติ จึงนับได้ว่าเป็นสิ่งที่จำเป็น

เครื่องควบคุมแบบตรรก เป็นเครื่องควบคุมที่ถูกนำมาประยุกต์ใช้ในระบบโรงงานอัตโนมัติ ทั้งนี้เพราะเครื่องควบคุมแต่ละชุด จะควบคุมเครื่องจักร เครื่องกล ที่ถูกจัดวางในที่ต่าง ๆ กัน จึงไม่เป็นการสะดวกที่จะนำคอมพิวเตอร์ มาทำการตรวจสอบสภาวะหรือส่งค่าตัวแปรให้กับเครื่องควบคุม ชุดต่อชุด เป็นการสิ้นเปลืองค่าใช้จ่าย และได้ประสิทธิภาพการทำงานที่ต่ำแต่ถ้านำเอาคอมพิวเตอร์ เพียงเครื่องเดียวแล้วสามารถใช้ส่วนต่อเชื่อมกับเครื่องควบคุม PLC จำนวน หลาย ๆ เครื่องได้ทั้งหมด อำนวยความสะดวกต่อการทำงานได้เป็นอย่างมาก เนื่องจากผู้ควบคุมสามารถตรวจสอบการทำงานของเครื่องจักรกลได้ทุกเครื่องที่มีเครื่องควบคุมการทำงานต่อในระบบนี้อยู่

จึงเป็นเหตุผลให้มีการพัฒนา โปรแกรมบนคอมพิวเตอร์ เพื่อให้เป็นโฮสคอมพิวเตอร์ กับเครื่องควบคุม จำนวนหลายชุดในระบบ ขณะเดียวกัน เครื่องควบคุมแต่ละเครื่องก็ยังคงทำงานเป็นอิสระต่อกัน

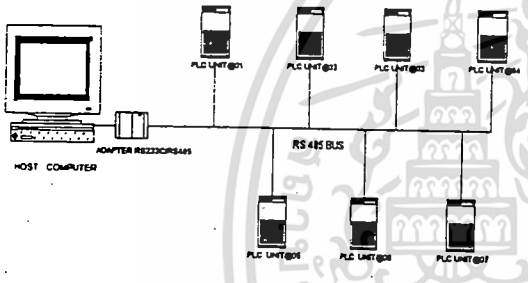
สำหรับงานวิจัยนี้ เป็นงานวิจัยต่อเนื่องจากบทความเรื่อง เครื่องควบคุมแบบตรรกที่โปรแกรมได้ ในการประชุมทางวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 18 ดังนั้น ฮาร์ดแวร์ ของเครื่องควบคุมที่ถูกพัฒนา จะไม่นำมากล่าวทั้งหมด แต่จะกล่าวในส่วนที่เกี่ยวข้องและพัฒนาเพิ่มเติมขึ้นมา คือส่วนของการสื่อสารข้อมูล ที่มีการต่อ ควบคุมแบบแยกกระจาย (MULTIDROP) และการออกแบบโปรแกรมการทำงานบนโฮสคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.โครงสร้างของระบบและข้อตกลงในการสื่อสารข้อมูล (PROTOCOL)**

*2.1ระบบทางด้านฮาร์ดแวร์ของระบบการต่อควบคุมแบบแยกกระจาย (MULTIDROP)*

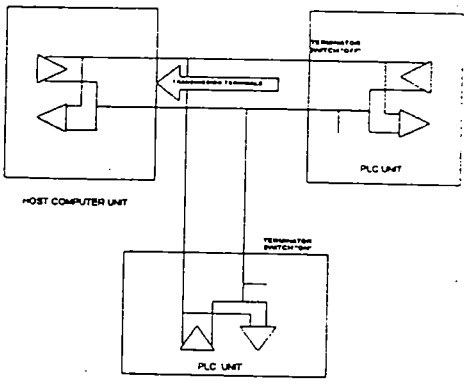
โดยต่อเป็นลักษณะ เครือข่ายกันระหว่างเครื่องควบคุมที่โปรแกรมได้ การสื่อสารข้อมูลกันภายใน จะถูกกระทำผ่านทาง พอร์ท อนุกรม เป็นแบบ RS485 ที่เชื่อมต่อแบบแยกกระจาย โดยเครื่องควบคุมแบบตรรก ถูกวางอยู่ในเครือข่ายได้ทั้งหมด 8 เครื่อง โดยแต่ละเครื่องจะมีตำแหน่งเฉพาะระบุประจำเครื่อง การกำหนดตำแหน่งกระทำได้โดย ปรับสวิทช์ (DIPSWITCHS) ได้ตั้งแต่ หมายเลข 0 ถึงหมายเลข 7 เพื่อวัตถุประสงค์ก็คือ การเรียกตรวจสอบสภาวะจากเครื่องควบคุมทั้งหมดได้ จากโฮสคอมพิวเตอร์ เครื่องเดียว ดังรูป 2-1




รูปที่2-1แสดงการเชื่อมต่อแบบแยกกระจาย(MULTIDROP)

**2.2 ระบบทางด้านฮาร์ดแวร์ เพื่อใช้ในการสื่อสารข้อมูล** หรือติดต่อระหว่าง โฮสคอมพิวเตอร์และเครื่องควบคุมจำนวนหลายจุด

ในระบบฮาร์ดแวร์เดิมมีการสื่อสารข้อมูล แบบอนุกรมตามมาตรฐาน RS232 C ในรูปแบบของสัญญาณแรงดัน จะมีข้อจำกัดอยู่มากคือจะต่อกับคอมพิวเตอร์เพื่อตรวจสอบสภาวะต่าง ๆ ได้ เพียงจุดต่อจุด เท่านั้น และระยะทางไม่ไกลนัก จึงได้ทำการปรับปรุงให้เป็นแบบ RS485 ที่ส่งสัญญาณในรูปแบบของกระแส ทำให้ลดปัญหาเรื่องระยะทางและจำนวนจุดต่อ เพราะการต่อเป็นแบบกระจาย (MULTIDROP)




รูปที่ 2-2 แสดงลักษณะการสื่อสารข้อมูลแบบอนุกรม RS485

**2.3 ข้อตกลงในการสื่อสารข้อมูล (PROTOCOL) กับอุปกรณ์ภายนอก**

ชุดของข้อมูลในการสื่อสารจะถูกเรียกว่าบล็อค บล็อคของข้อมูลจะถูกส่งจากเครื่องคอมพิวเตอร์ (เครื่องคอมพิวเตอร์ต่อไปจะเรียกว่า HOST) ไปในระบบการเชื่อมต่อ ซึ่งจะเรียกว่าบล็อคคำสั่ง (COMMAND BLOCK) และบล็อคของข้อมูลที่ถูกส่งจากระบบการเชื่อมต่อไปสู่ HOST จะเรียกว่า บล็อคตอบสนอง(RESPONSE BLOCK) ในระบบการเชื่อมต่อสื่อสารแบบหลายจุด แต่ละบล็อค ไม่ว่าจะเป็นบล็อคคำสั่งหรือบล็อคตอบสนองก็ตาม จะเริ่มต้นด้วย อักขระ "@" ตามด้วยตำแหน่งเฉพาะ (UNIT NUMBER) ตามด้วยคำสั่ง (HEADER) ข้อมูล (DATA) และ อักขระ "*" สิ้นสุดด้วยรหัสกำกับบล็อค (FRAME CHECK SCQUENCE CODE : FCS) และรหัสปิดท้ายบล็อคที่เป็นอักขระ [CR]

*รูปแบบของบล็อค (BLOCK FORMAT)*

| @ | X | X | X | X | DATA | * | X | X | CR |
|---|---|---|---|---|---|---|---|---|---|
| (ตำแหน่ง) | (คำสั่ง) | | | | | | FCS | | ปิด |

ช่วงคำนวณรหัสกำกับบล็อก

จำนวนอักขระในแต่ละบล็อคทั้งหมดจะต้องไม่เกิน 128 ตัว และช่วงของการคำนวณเพื่อหา รหัสกำกับบล็อคจะอยู่ระหว่างอักขระเริ่มต้น (@)ไปจนถึงสิ้นสุด DATA ที่ เป็น "*"

*การคำนวณหารหัสกำกับบล็อค (FCS CALCULATION)*

FCS เป็นข้อมูลขนาด 8 บิท ที่ถูกแปลงไปเป็นอักขระ ASCII และ 8 บิทดังกล่าวนี้เป็นผลลัพธ์จากการ EXCLUSIVE OR [XOR] กันเป็นลำดับของ แต่ละอักขระ ในบล็อค เริ่ม จากอักขระแรกจนถึงอักขระสุดท้ายของDATAเช่น {@01RL00000 010*74 [CR] }

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายคือ อ่านพื้นที่ LINK RELAY เริ่มที่ตำแหน่ง 0000 อ่านมาจำนวน 10 ตำแหน่ง

| อักขระ | รหัส(ASCII BINARY) (HEX) | | | XOR (HEX) |
|---|---|---|---|---|
| @ | 0100 | 0000 | [40] | |
| 0 | 0011 | 0000 | [30] | 112 [70] |
| 1 | 0011 | 0001 | [31] | 65 [41] |
| R | 0101 | 0010 | [52] | 19 [13] |
| L | 0100 | 1100 | [4C] | 95 [5F] |
| 0 | 0011 | 0000 | [30] | 111 [6F] |
| 0 | 0011 | 0000 | [30] | 95[5F] |
| 0 | 0011 | 0000 | [30] | 111[6F] |
| 0 | 0011 | 0000 | [30] | 95 [5F] |
| 0 | 0011 | 0000 | [30] | 111 [6F] |
| 0 | 0011 | 0000 | [30] | 95 [5F] |
| 1 | 0011 | 0001 | [31] | 110 [6E] |
| 0 | 0011 | 0000 | [30] | 94 [5E] |
| * | 0010 | 1010 | [2A] | 116 [74] |

*ตัวอย่างโปรแกรมในการคำนวณรหัสกำกับบล็อค*

```
Sub fcs()
        Dim a,b,c
        a = "@ 01 RL 00000001*"
        b = Len(a)
        c = 0
        for b= 0 to b - 1
           c = c Xor Asc (Mid (a,b+1,1))
        Next b
        fcs = HeX (c)
end Sub
```

*คำสั่งและคำตอบสนอง*

ในขณะที่ทำการเชื่อมต่อเพื่อสื่อสารข้อมูล HOST สามารถที่จะทำการเฝ้ามองการดำเนินการและสามารถที่จะทำการควบคุมเครื่องที่โปรแกรมได้ ถ้าอยู่ในสถานะการเฝ้ามอง HOST จะต้องส่งคำสั่งไปถามตามชนิดของข้อมูลที่ต้องการ แต่เครื่องควบคุม หรือถ้าในสถานะการควบคุมก็สมารถที่จะส่งคำสั่งไปทำการเปลี่ยนแปลงค่าของข้อมูลที่อยู่ในหน่วยความจำโดยตรง เช่น ข้อมูลของ อินพุท/เอาท์พุท เป็นต้น

เวลาของการตอบสนองจะแปรไปขึ้นอยู่กับความเร็วในการส่งผ่านข้อมูล จำนวนของข้อมูล และเวลาในการสแกนของเครื่องควบคุม และถ้าเวลาในการสื่อสารมากขึ้นก็เป็นผลให้เวลาในการสแกนมากขึ้นตามไปด้วย ต่อไปจะเป็นคำสั่งต่าง ๆ ที่เป็นข้อกำหนด (Protocol) ที่ใช้ในการสื่อสาร

*บล็อกคำสั่งประกาศ*

บล็อคคำสั่งประกาศถูกส่งโดย HOST ไปเพื่อตรวจสอบว่ามีเครื่องควบคุมอยู่ในการเชื่อม ต่อหรือในเครือข่ายหรือไม่

รูปแบบคำสั่ง " @[Unit Number]AT*[FCS][CR] "

เช่น BLOCK COMMAND เป็น "@01AT*7E[CR]" หมายถึง HOST ส่งออกไปเพื่อ ประกาศ ถามถึงเครื่องควบคุมที่อยู่ในการเชื่อมต่อมีตำแหน่งเฉพาะที่ 01 หรือไม่ ถ้าเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมมีจริงและได้รับคำประกาศ ดังกล่าวก็จะส่งคำตอบสนองออกไป ดังนี้

รูปแบบคำตอบสนอง "@[Unit Number]PC84*[FCS][CR] "

เช่น BlockResponseเป็น "@01PC84*74[CR]" หมายถึงเครื่องควบคุมตำแหน่ง เฉพาะที่ 01เท่านั้นที่ตอบออกไป ซึ่งจะแสดงให้ HOST ทราบว่าเครื่องควบคุมที่ตำแหน่ง 01 ในระบบการเชื่อมต่อยังคงทำงานปกติ

*บล็อกคำสั่งอ่านพื้นที่ Input/Outputt/Internal Relay*

รูปแบบคำสั่ง " @[Unit Number]RI[ตำแหน่งเริ่มต้น][จำนวนข้อมูล]*[FCS][CR] "

เช่นถ้า HOST ต้องการอ่านพื้นที่ของ I/0 หรือ Internal Relay ของเครื่องควบคุมที่วงอยู่ ใน ระบบการเชื่อมต่อ ตำแหน่งเฉพาะที่ 02 ตำแหน่ง I/0 ที่ 0010 มาจำนวน 5 ตำแหน่ง ก็สามารถจัด BlockCommand ได้ดังนี้ "@02RI00100005"XX[CR]" (FCS XX : ASCII 2 digit)

รูปแบบคำตอบสนอง " @[Unit Number]RI[ตำแหน่งเริ่มต้น][ข้อมูลXX(1)][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR] "

เช่น ถ้าคำตอบสนอง ของเครื่องควบคุม ในตำแหน่งเฉพาะ ที่02ตอบมาเป็น "@02RI0010234455566AB*XX[CR]" ก็หมายความว่า ตำแหน่งหน่วย ความจำของ I/O หรือ Internal Relay ที่ 0010 มีข้อมูล 23,44,55,66,AB [ฐาน16] ตามลำดับ

*บล็อกคำสั่งเขียนพื้นที่ Input/Output/Internal Relay*

รูปแบบคำสั่ง "@[Unit Number]WI[ตำแหน่งเริ่มต้น][ข้อมูลXX(1)][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เช่นต้องการเขียนข้อมูลลงในตำแหน่ง I/0 หรือ Internal Relay ของเครื่องควบคุมที่ตำแหน่งเฉพาะ 03 ในตำแหน่งที่ 0010 ด้วยข้อมูล 12,34,56,78,9A จำนวน 5 ตำแหน่ง ก็สามารถจัด BlockCommand ได้ดังนี้ "@03WI00101234567 89A*XX[CR]"

รูปแบบคำตอบสนอง" @[Unit Number]WI[รหัสตอบสนอง]*[FCS][CR]"

(XX :Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@03WI00*XX[CR]"

*บล็อกคำสั่งอ่านค่าเป้าหมาย Timer*

เพื่อสามารถให้ Host รู้ค่าเป้าหมาย (Set Value) ที่ได้ตั้งไว้ที่ตัวเวลา

รูปแบบคำสั่ง "@[Unit Number]RT[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR] "

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวตั้งเวลาที่ 12 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด BlockCommand ได้ดังนี้ "@04RT00120001*XX[CR]"

รูปแบบคำตอบสนอง "@[Unit Number]RT[ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้ค่า ตอบสนองออกมาสมมุติว่าค่าเป้าหมายของตัวตั้งเวลาตำแหน่งที่ 12 มีค่า #0150 และ Block Response เป็น "@04RT00120150*XX[CR] "

*บล็อกคำสั่งอ่านค่าเป้าหมาย Counter*

เพื่อสามารถให้ Host รู้ค่าเป้าหมาย (Set Value) ที่ได้ตั้งไว้ที่ตัวตั้งนับ

รูปแบบคำสั่ง "@[Unit Number]RC[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR]

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวนับที่ 10 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด BlockCommand ได้ดังนี้ "@04RT00100001*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit Number]RC[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR] "

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวนับตำแหน่งที่ 10 มีค่า #8000 และ BlockResponse เป็น "@04RT0010800*XX[CR]"

*บล็อกคำสั่งเขียนค่าเป้าหมาย Timer*

เพื่อที่ Host สามารถตั้ง แก้ไขค่าเป้าหมายใหม่ของตัวตั้งเวลาแก่เครื่องควบคุมที่อยู่ในระบบเชื่อมต่อได้ตามต้องการ

รูปแบบคำสั่ง"@[Unit Number]WT[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นถ้าต้องการตั้งค่าเป้าหมายแก่ตัวตั้งเวลาของเครื่องควบคุมที่วางในระบบเชื่อมต่อที่มีตำแหน่งเฉพาะที่ 05 และตัวตั้งเวลาตำแหน่งที่ 00 ด้วยค่าเวลา #0200 สามารถจัด BlockCommand ได้ดังนี้ "@05WT0000200*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit Number]WT[รหัสตอบสนองXX]*[FCS][CR]"

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@05WT00*XX[CR]"

*บล็อกคำสั่งเขียนค่าเป้าหมาย Counter*

ลักษณะของข้อกำหนดจะเหมือนกับเขียนค่าเป้าหมายให้กับตัวตั้งเวลา ต่างกันที่ Header

รูปแบบคำสั่ง"@[Unit Number]WC[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เช่นถ้าต้องการตั้งค่าเป้าหมายแก่ตัวนับของเครื่องควบคุมที่วางในระบบเชื่อมต่อที่มีตำแหน่งเฉพาะที่ 05 และตัวตั้งเวลาตำแหน่งที่ 00 ด้วยคำนับ #0200 สามารถจัด BlockCommand ได้ดังนี้ "@05WC00000200*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit number]WC[รหัสตอบสนองXX]*[FCS][CR]"

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@05WC00*XX[CR]"

*บล็อกคำสั่งอ่านค่าปัจจุบัน Timer*

เพื่อที่ Host สามารถที่จะทำการเฝ้ามองความเป็นไปของค่าเวลาที่กำลังทำงานของตัวตั้งเวลาในเครื่องควบคุมตำแหน่งเฉพาะใด ๆ ที่วางในระบบเชื่อมต่อได้

รูปแบบคำสั่ง"@[Unit Number]PT[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR]

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวตั้งเวลาที่ 12 และ 13 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด BlockCommand ได้ดังนี้ "@04PT00120002*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit Number]PT[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวตั้งเวลาตำแหน่งที่ 12 มีค่า #0150 และตัวตั้งเวลาตำแหน่งที่ 13 มีค่า #0200 BlockResponse เป็น "@04rt001201500200*XX[CR]"

*อ่านค่าปัจจุบัน Counter*

รูปแบบคำสั่ง"@[Unit Number]PC[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR]"

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวนับตำแหน่งที่ 10 มีค่า #8000 และ BlockResponse เป็น "@04pt00108000*XX[CR]"

*อ่านจากพื้นที่โปรแกรม UpLoad Program*

เพื่อให้ Host สามารถที่จะนำข้อมูลส่วนโปรแกรมบูลลีนที่ผู้ใช้ได้ทำการโปรแกรมไว้ในหน่วยความจำนั้นขึ้นมาที่ Host เพื่อทำการเก็บรักษาไว้

รูปแบบคำสั่ง"@[Unit Number]UL*[FCS][CR]"

เช่นถ้า Host ต้องการนำโปรแกรมส่วนบูลลีนจากเครื่องควบคุมที่มีตำแหน่งเฉพาะเป็น 01 มาจากจุดเริ่มโปรแกรมถึงคำสั่งสุดท้าย (END) มาสามารถจัด BlockCommand ได้เป็น "@01UL*72[CR]" และเครื่องควบคุมที่ตำแหน่งเฉพาะ 01 จะให้คำตอบสนองออกมาเป็นชุดข้อมูลที่ละ 20 Bytes เริ่มที่ตำแหน่งในหน่วยความจำที่ 8000 และชุดต่อไปของข้อมูลจะมีตำแหน่งเริ่มต้นที่ สอดคล้องกับตำแหน่งที่ถูกถ่ายเทขึ้นมาด้วย หรือจนกว่าจะพบคำสั่ง END ดังรูปแบบดังต่อไปนี้ ข้อสังเกต รูปแบบคำตอบสนองนั้นจะมีลักษณะเหมือนกับคำสั่งในการเขียนลงพื้นที่โปรแกรม ทั้งนี้ก็เพื่อที่จะ ให้สามารถนำชุดข้อมูลดังกล่าวนั้นเขียนลงไปใหม่ที่ส่วนโปรแกรมได้เลยโดยไม่ต้องเปลี่ยนแปลงรูปแบบ

รูปแบบคำตอบสนอง"@[Unit Number]UL[ตำแหน่งเริ่ม8000][ข้อมูลXX(..)][ข้อมูลXX(20)]*[FCS][CR]"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขียนลงพื้นที่โปรแกรม DownLoad Program เพื่อให้ Host สามารถทำการส่งผ่านข้อมูลส่วนโปรแกรมบูลลีนเข้าไปไว้ในพื้นที่โปรแกรมของเครื่องควบคุมที่มีตำแหน่งเฉพาะนั้น ๆ ได้

รูปแบบคำสั่ง "@[Unit Number]DL[ตำแหน่งเริ่ม8000][ข้อมูลXX(..)][ข้อมูลXX(20)]*[FCS][CR]"

ข้อมูลของโปรแกรมบูลลีนที่ต้องการเขียนลงในหน่วยความจำส่วนโปรแกรมนั้นก็จะต้องไม่มากกว่า 20 Bytes เช่นเดียวกัน และตำแหน่งที่ต้องการวางลงในหน่วยความจำก็สามารถกำหนดได้ (ตำแหน่งในหน่วยความจำส่วนโปรแกรมบูลลีนเริ่มที่ 8000)

รูปแบบคำตอบสนอง"@[Unit Number]DL[รหัสตอบสนองXX]*[FCS][CR]"

(XX : Response Code)

00 = Data Complete

08 = Data Error

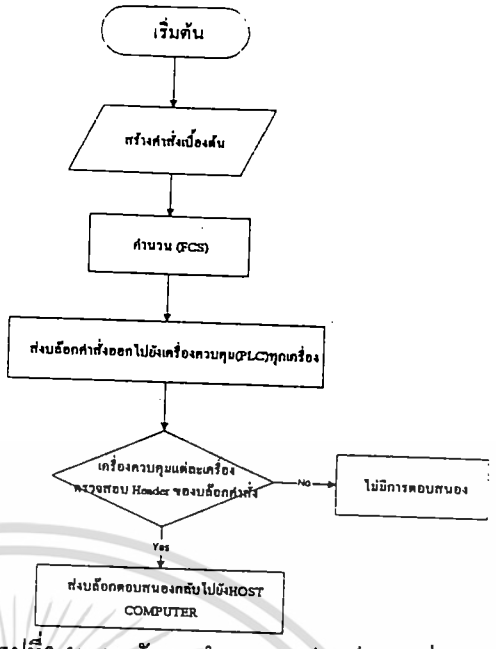
และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@01DL00*XX[CR]"

**3)แนวคิดและการออกแบบโปรแกรม**

ในการเขียนโปรแกรมคำสั่งบูลลีนลงในเครื่องควบคุม PLC จะประสบปัญหาในเรื่องของการจัดเก็บข้อมูลลงในหน่วยความจำสำรอง และการตรวจสอบสภาวะการทำงาน คือไม่สามารถตรวจตรวจสอบสภาวะของหน่วยความจำต่าง ๆ ได้ครั้งละหลาย ๆ ตำแหน่งเพื่อแก้ไขปัญหาจึงได้ออกแบบโปรแกรมเพื่อรองรับความต้องการ โดยจัดแบ่งการติดต่อสื่อสารระหว่างเครื่องควบคุมกับโฮสคอมพิวเตอร์ ซึ่งการออกแบบจะได้แสดงดังต่อไปนี้

*3.1 การส่งบล็อคคำสั่งจากโฮสคอมพิวเตอร์ไปยังเครื่องควบคุม PLC*

โฮสคอมพิวเตอร์จะส่งบล็อคคำสั่งออกไปทางพอร์ทอนุกรมของคอมพิวเตอร์ ผ่านเครื่องแปลงระบบ RS232 เป็น RS 485 ไปยังเครื่องควบคุม PLCแต่ละเครื่อง เครื่องควบคุมจนตรวจสอบหัวของคำสั่ง หรือ หมายเลขประจำเครื่อง ถ้าตรงกันก็จะส่งบล็อคตอบสนองกลับไปยังโฮสคอมพิวเตอร์ซึ่งหัวของคำสั่ง (Header) นี้กำหนดได้จากโปรแกรมบนคอมพิวเตอร์




รูปที่3-1แสดงผังการทำงานการส่งบล็อคคำสั่งโฮสคอมพิวเตอร์

*3.2 การรับบล็อคตอบสนองจากเครื่องควบคุม (PLC) ของโฮสคอมพิวเตอร์ เพื่อนำมาแสดงผล*

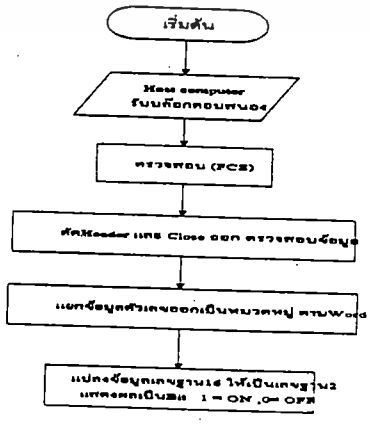
หลังจากที่ส่งบล็อคคำสั่ง เครื่องควบคุม (PLC) ที่มีหมายเลขระบุตำแหน่ง ตรงกันก็จะส่ง บล็อคตอบสนอง กลับมา ยกตัวอย่างการตรวจสอบสภาวะของ OUTPUT

| @ | X | X | X | X | DATA | * | X | X | CR |
|---|---|---|---|---|---|---|---|---|---|
| (ตำแหน่ง) | | (คำสั่ง) | | | | FCS | | ปิด | |

โปรแกรมจะตรวจสอบจนได้ข้อมูลที่ถูกต้องแล้ว ทำการ ตัดส่วนหัว [@,UNIT NO, COMMAND] และส่วนท้าย [*,FCS) ออกเหลือเพียงส่วนที่เป็นข้อมูล (DATA)

ส่วนข้อมูล (DATA) จะประกอบด้วย WORD ที่เริ่มต้น และตามด้วยข้อมูลที่เป็นเลขฐาน 16 WORD 2 หลัก (8 bit) จากนั้นก็นำข้อมูลมาแปลงเป็นเลขฐาน 2 แสดงผลออกเป็น bit ในแต่ละ word ซึ่งพอจะสรุปเป็นผังการทำงานดังนี้

รูปที่ 3 - 2 แสดงผังการทำงานของการแสดงผลข้อมูล

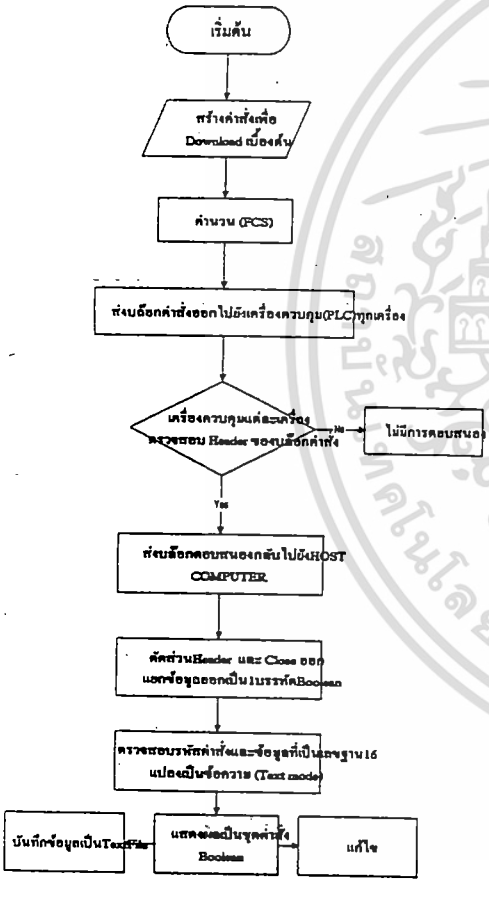



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

*3.3 การนำคำสั่งโปรแกรมบูลลีนจากเครื่องควบคุม (PLC) มาแก้ไขหรือบันทึกลงบนโฮสคอมพิวเตอร์*

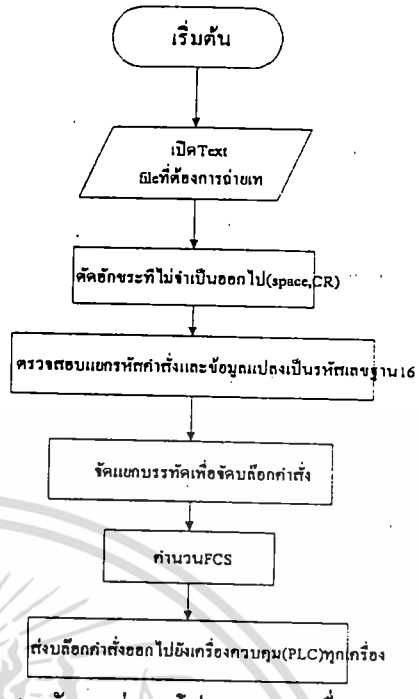
คือการส่งบล็อคคำสั่งเพื่อให้ PLC ส่งข้อมูลที่เป็นรหัสเลขฐาน 16 มายังโฮสคอมพิวเตอร์ จากนั้นโฮสคอมพิวเตอร์มีหน้าที่แปลงรหัสต่าง ๆ ให้เป็นข้อความ (Text mode) เพื่อแสดงผลบนหน้าจอ และสามารถที่จะแก้ไขหรือบันทึกข้อมูลต่าง ๆ เป็น แบบ Text File

ตัวอย่างการแปลงเลขรหัสเลขฐาน 16 ให้เป็นข้อความเช่น "80 0001"จะแยกออกเป็น 2 ส่วนคือ 2 ไบท์ แรก(80) เป็นคำสั่ง 4 ไบท์ ที่เหลือเป็นข้อมูล เมื่อ 80 = LD จะได้ว่า LD 0001 สำหรับรหัสต่าง ๆ ของคำสั่งบูลลีนนั้นอ้างอิงตามเอกสารอ้างอิงหมายเลข[1]




รูปที่ 3 - 3 แสดงผังการแปลงข้อมูลจาก PLC เป็นข้อความ

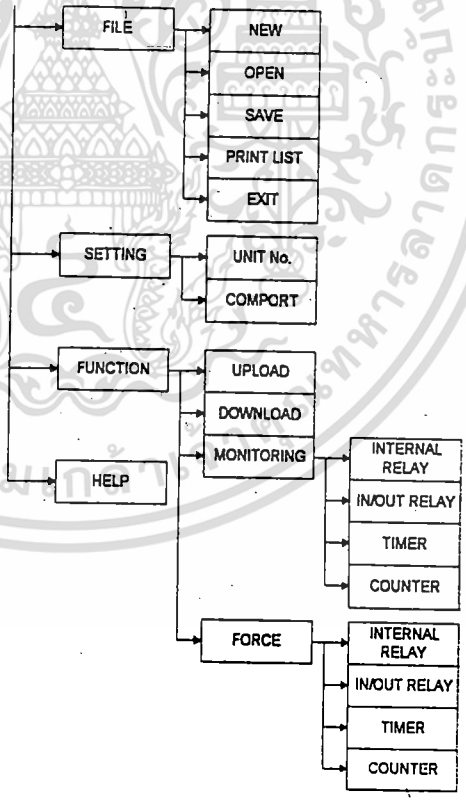
*3.4 การนำโปรแกรมบูลลีนที่บันทึกไว้บนคอมพิวเตอร์ เพื่อถ่ายเทให้กับเครื่องควบคุม (PLC)*




รูปที่ 3 - 4 แสดงผังการถ่ายเทโปรแกรมลงเครื่องควบคุม PLC

**4 ผลการทดลอง**

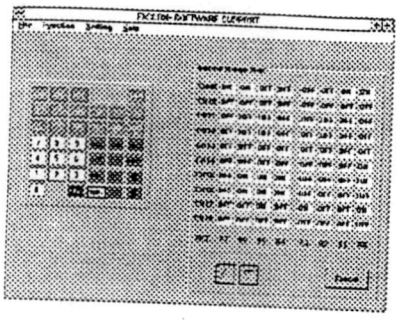
*4.1 แผนผังของโปรแกรมที่ออกแบบ*



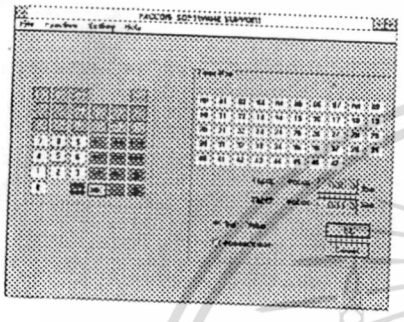

รูปที่ 4-1 แสดงแผนผังของโปรแกรม

*4.2 การตรวจสอบสภาวะของ รีเลย์ ตัวตั้งเวลา และตัวนับ*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

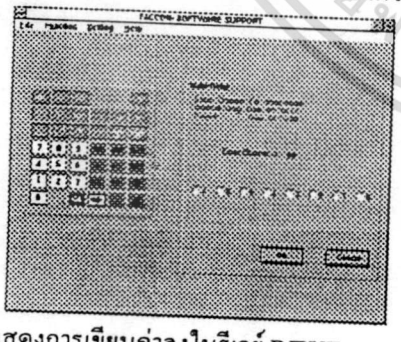


รูปที่ 4-2 สดงการตรวจสอบสภาวะของรีเลย์ภายใน



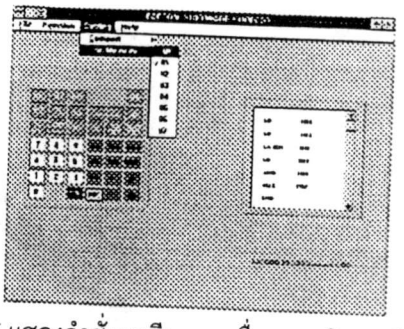
รูปที่ 4-3 แสดงการตรวจสอบ สภาวะของตัวตั้งเวลา

รูปที่ 4-4 แสดงการตรวจสอบสภาวะของตัวนับ



รูปที่ 4-5 แสดงการเขียนค่าลงในรีเลย์ INPUT และ OUTPUT

*4.3 การแสดงโปรแกรมคำสั่งบูลลีนที่รับมาจากเครื่องคุม*



รูปที่4-6 แสดงคำสั่งบูลลีนบนเครื่องคอมพิวเตอร์

5. บทสรุป

จากการศึกษาและทดลองใช้งาน ให้ผลการทำงานเป็นที่น่าพอใจในระดับหนึ่ง และมีแนวโน้มที่จะพัฒนาประสิทธิภาพของโปรแกรมให้สูงขึ้นต่อไปเพื่อความสะดวกในการเขียนโปรแกรมลงบนเครื่องควบคุม PLC เช่นเขียนโปรแกรมเป็นบล็อกการทำงานแล้ว แปลงเป็นชุดคำสั่งบูลลีนเป็นต้น

โปรแกรมนี้ใช้ได้เฉพาะกับฮาร์ดแวร์ที่สร้างขึ้นเท่านั้น แต่สามารถนำหลักการนี้ไป ใช้กับฮาร์ดแวร์อื่นได้ไม่ยาก

**เอกสารอ้างอิง**

[1] วิริยะ กองรัตน์, ผ ศ. วิทยา ทิพย์สวรรณพร, ผ ศ. สุพรรณ กุลพาณิชย์, " เครื่องควบคุมแบบตรรกที่โปรแกรมได้ " ,การประชุมวิชาการทางวิศวกรรมทางไฟฟ้า ครั้งที่ 18 , มหาวิทยาลัยเทคโนโลยีมหานคร , 1995

[2] Perter Hamann, Steve Willings " MINI - H-TYPE PROGRAMMABLE LOGIC CONTROLLERS OPERATION MANUAL " , July 1990

[3] Joe Campbell , C Programmer ' s Guide to Serial Communications Second Edition , Sams Publishing, 1994

[4] Paul Bonner , PC MAGAZINE VISUAL BASIC UTILITYS , Ziff Daus press emeru ville, California , 1993

[5] Mark S. Burgess , ADVANCED VISUAL BASIC , Addison - Wesley Publishing Co. ltd., 1994

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้